Oral History of Dr. Suhas Patil

Interviewed by:
Douglas Fairbairn

Recorded: August 2, 2010
Mountain View, California

CHM Reference number: X5664.2010

Doug Fairbairn: I'm Doug Fairbairn. I'm interviewing Dr. Suhas Patil and the first thing that we'd like to get as far as Suhas' comments here today is since he is familiar with the Computer History Museum and has been involved with it for some time, actually from the beginning, I would like to have him comment on what he thinks the most important contribution and the greatest value that the Computer History Museum can bring to the industry and to this area. So if you'll just comment on what you think the values that the Computer History Museum brings to out community and to our industry.

Dr. Suhas Patil: Well we have witnessed and are still witnessing some of the fastest and biggest changes in technology, you know, for the last 100 years, and certainly for the last 50 years we have seen many of these changes being led by advances in computers. Now the computing history is a lot longer than just that of a century. You know we in Computer History Museum we were going to have something saying "Two thousand years of computer history," okay? Dramatic change started towards end of 1950s and it's been amazing. So we are in it. Things are happening and for the future generation we need to capture what is going on close enough to when it is taking place so that at least we preserve it. Once time passes it will be nearly impossible to collect this, and there is much to be preserved including what was happening, how it was happening, so that our future generations would have the material to study and understand and put to use in their own life. So that is a function of preservation, okay? Another purpose is to bring people together who have participated so that they might pass on the knowledge to other folks here. So it's both a museum as well an educational institution. Hardware is much easier to see, but there's much to be learned and understood in the software and algorithms and all of that is in the Computer History Museum. It's preserving. And subsequently we need to be a museum that is not just bound by the physical size of the museum but we need to reach people through cyberspace. So all of these things have to be done. There's no better place than Silicon Valley to do that.

Fairbairn: Okay excellent. I think that's great insight. Now I'd like to pose to you a couple questions that, given your background in technology and your witnessing some of these great changes over the last 30, 40, 50 years, and we'd like to get your insight and opinions to help and guide and influence future generations. So the first of those questions is what problems do you think the technology industry can solve for society in the future?

Patil: Well you know technology is a means to solving problems, and towards that at different times you face different problems and technologies can be skillfully applied to that. So say for example today we have problems of energy and reducing our consumption of energy. I think much can be done so that we get more use from the energy we spend, and certainly much can be done from computer science or use of computers and information technology. Smart grid is just beginning and energy efficiency of the building. Besides the material sciences contribution, I think we can do a lot. That's one example of it. When you look broadly at our chemical factories, they are really very crude by standards of what they can be and the more we make them efficient the less waste we will make and the less energy we will use. There's another very interesting example. The field of biosciences has been kind of building up steam. Let me point out that transistors and computers came into existence about 50 years after structure of the atom was discovered. Well double helix was discovered in mid of last century. Fifty years have passed. It's now time for good things to come from there. Medicine is about to change fully and medicine does rely very much on technology to do that. So we can see continuous improvement. The Internet is like electrification. Its impact is phenomenal. It has been helped by the semiconductor industry. It's helped by software, helped by systems. So we're going to see clearly another century of very, very rapid application of technology, tools, to the needs of the human being. Beyond that, it's very hard for me to assess. <laughs>

Fairbairn: So given this tremendous opportunity for application of technology, what advice would you give a young person starting out today in terms of perhaps pursuing a career in technology or just in general? What general advice would you lend?

Patil: You know the first observation I would like to pass on is in the generation prior to mine, when a person finished education or got his job pretty much worked in one place all his life. In my own generation we've done multiple things. And the generations to come we will be changing jobs and applying ourselves in different fields. Therefore it is very important to have a foundation so as to have flexibility. Good grounding is absolutely a must, and this grounding occurs in the K to 12 but you are very much aware as an individual and working on your own as opposed to parents starting from somewhere in high school and college. That is a time. Take the courses and study, study for learning's sake things that you may not ever have thought you wanted to learn. Don't ignore humanities if you are studying science. And those who are studying arts, science is not a mystery. If you want to learn, math is not a mystery. Take those courses and learn.

Fairbairn: Excellent. I think that's superb advice. Okay let's move on to the major topic at hand and that is your own background and experiences and things that influenced your life and impact that you've had on the industry over the past 40, 50 years. And so to begin just as you were talking about foundation, I'd like to sort of go back to your foundation, your family, where you were born, the community you grew up in, and just tell me a little bit about that. You were born in India, I understand. Tell me about your family, the environment, the community that you were in and how that that steered you in the directions that you took in terms of a technology career. So just go back to the beginning and tell me about your parents maybe your siblings or whatever in the context of how it helped steer you.

Patil: My great grandfather was just purely a farmer. My grandfather was a farmer, a teacher, and a lawyer. They were all from a small place and he had great confidence in his children both sons and daughters. There were eight siblings, my father was the oldest and he inculcated in my father love for learning. He was the first one from his small town to go to a university to get an engineering degree.

Fairbairn: Your father was?

Patil: Yes. And a whole bunch of people followed after that. So he studied at the best technical university in India at that time, but not as Hindu University and he ended up to work in Tata Steel Company. His own background was electrical engineer, mechanical engineer, and by training metallurgist at the company and finally they borrowed him in accounts department because he could do cost accounting better than others. And they never let him go back to the metallurgical labs. So with that background and growing up in Jamshedpur, which is where the Tata Iron and Steel Company was, innovation was so obvious. Where I grew up, 50 years before that there was nothing but jungles. And here it was; we were very aware that someone had a dream and a passion to spend the energy and talk people into helping them build the biggest steel company in India. Now those days such companies had to build everything including the school, the hospital, you know the waterworks and so on.

Fairbairn: So was this very isolated from any major city within India?

Patil: It was about 170 miles west from Calcutta which was the closest. So it was not lost on all young people including me that you could do things. We were much inspired and that was a very, very key for me. The other key was that my father was an engineer, but to earn extra money to help his siblings go to college he had a radio repair shop and by the time I became teenager he was no longer doing that, but my mother had kept all the instruments including old radios and World War II gear junk, but it was great source for valves and capacitors, resistors, and I learned my electronics from the ham radio's manual, and I learned English to read *Popular Science*.

Fairbairn: Were you taking English in school as well?

Patil: Yes, yes but my real motivation in learning English was to be able to read *Popular Science* which my father was getting, and my father knowing electronics was- I had a teacher at home so that helped out. So by the time I finished high school I actually was very good in debugging radios, fixing them. I'd already built circuitry, built receiver, built transmitter, all kinds of things so I had experimented so I had a very good foundation. That developed my interest in going into electronics. I wanted to be a scientist first, not an engineer. Later on you know I actually focused myself in engineering.

Fairbairn: So then you went on to university.

Patil: Yes.

Fairbairn: Tell me about that, how you chose that. Was that an automatic or how did that come about?

Patil: There was nothing automatic. <laughs> I went to study intermediate science after my high school and that was St. Xavier's College in Calcutta. Two years of that and that was a science education. If I had stayed there I would have gotten Bachelors of Science. I graduated with intermediate science and I wanted to go to the IIT which was in between my hometown and Calcutta. Indian Institute of Technology Kharagpur. And that's a very competitive entrance examination and I did well. And electronics was not popular at that time. Mechanical engineering and civil engineering were popular, but they had a department of electronics and communication. And I said "This is the department for me." I had only 12 students in my class. Electrical engineering had 100 so you can now imagine, it's almost like having a private tutor Today there are a lot of students.

Fairbairn: And this was the late 1950s?

Patil: No '60s, early '60s yeah.

Fairbairn: And how long was that and was there any notable events or professors or people that happened, people that you met then that influenced your future career?

Patil: Absolutely. So the professors I had from my regular education, many of them studied abroad. And were very top notch. In particular a certain Professor who taught me microwave and circuits and things like that. It was phenomenal. And we also had the professor of communications so these were very, very top-notch people. And they inspired us to learn very thoroughly. And the way things were, if we hadn't, there was no place for us to hide. Of course but we did want to learn and so I had a fantastic education. And in my last year of my Bachelor's education Professor Don Bitzer had come there as a visiting professor. He was from University of Illinois. So he and I became very good friends and I learned a lot from him particularly about transistors and he had brought a bunch of transistors with him. I used those transistors to build my project. This was in 1964, '65 when I built a counter <laughs> three decimal digit counter which if we had been an entrepreneur at that time I could have just built it as a product, it was that good.

Fairbairn: You could have sold that, huh?

Patil: <laughs> So with his presence as well as those who had gone abroad to study I was very aware of American Universities. And IIT Kharagpur was modeled after MIT education and the books I studied from, many of them were written by American professors and I had access to radiation series, learned about transistors through this what is that U.S. magazine, not *Science.*

Fairbairn: *Scientific American.*

Patil: *Scientific American*, and so I had a good understanding and I had at that time said I would like to do doctoral work and my ideal place was MIT because I both respected and I was very aware it was a very dynamic place. What I didn't know whether anybody would give me admission. On top of that, I would need some kind of assistance because my father could not have sent me abroad for education. And fortunately when I applied I received both and that was fantastic.

Fairbairn: Were there options for you in India? Did you consider staying in India? Or when you decided to pursue a doctorate degree was going to America at that time sort of the obvious-

Patil: Well going to America for doing doctorate was a dream so that would have been my first preference. If that had not happened I probably would have gone to work at TIFR which is a research institute Tata Fundamental Research Institute, that would have been my second preference. And I'd done well in my studies. I was always considered a good student but did not perform at the top of the class in the marks as I entered IIT. Okay, I don't know why, but that's how it was so everybody has expectation was bigger than my actual achievement in terms of grades. But I must have been doing something right because by the time I finished I was number one. So I gained much from being at IIT. So having done well academically I had a wide-open choice. And if nobody would have hired me, I had enough skills that could have started a business, maybe starting with building electronic things or-

Fairbairn: Selling that counter that you designed, right?

Patil: <laughs> Yeah. So I was not worried at all whatsoever.

Fairbairn: So were there other students who from IIT at that time who went to MIT as well or were you the only one at that time?

Patil: People had already come before me and even in my batch at MIT there were two other IIT students who had come in the same department. So there were not very many people but there was sufficient number of people who had come.

Fairbairn: So when you went to MIT was this the first time you had traveled abroad outside of India?

Patil: Absolutely. First time I had traveled on a plane. First time I had traveled abroad.

Fairbairn: So how did you find that transition? Were there some funny stories coming into arriving at MIT and the student who had never left-

Patil: I'll tell you my funniest sort of experience or lack thereof. I had read all these comics and I was fully expecting to meet red Indians <laughs> when I arrived to America. To my great disappointment I didn't find any on Massachusetts Avenue. So that was my first impression. Second I said "Wow, this nice looking cars lighting up all the streets! This looks like a showcase." And I was very impressed with the physical infrastructure, and of course I got to meet the professors whose books I had studied from and that was phenomenal. I was very lucky I have to tell you. I was a TA for Amar Bose in my first semester and a teaching assistant for Paul Gray the second semester. I couldn't have asked for better people to work under and have as my mentors.

Fairbairn: So you entered MIT. Your path was clear at the time? What particular area did you want to pursue at MIT and was that how things worked out? Tell me about your undergraduate career there.

Patil: So I went to graduate school at MIT.

Fairbairn: Sorry, your graduate.

Patil: I had by the time I went to MIT it was clear in my mind I wanted to study computer science and learn to build computers, that was my interest. And the first year I had just spent studying and preparing for qualifying examination and at the end of the year I approached Professor Dennis who was in Project MAC to see if I could be his research assistant and be in the Computation Structures group.

Fairbairn: Describe Project MAC briefly, what does that stand for?

Patil: Project MAC was the Multi-Access Computer. As you know from historical records, here the first time-sharing system was developed at MIT. It was called CTSS and even at that time, now we are speaking about mid sixties, there were discussions going on and visualizing how computing could be a utility. Can you imagine that? That's good forty-

Fairbairn: Good foresight.

Patil: Forty years ago, right? So they're working on what would it take? How might it be done? And there were vigorous debates about it, and the mainframes at that time were not quite properly designed for it. This is the learning MIT researchers had from having built the compatible time-sharing system, CTSS, right? So this Multi- MAC, Multi-Access Computer this was a very massive initiative and it was funded by DARPA. At that point it was a big effort in whose context lots of things were done and many things came from it which we have seen benefits of. The ARPANET was associated with the same effort and so on. And Multics became sort of a very big computer and Bell Lab researchers said "Well we'd like to have a lighter one or a different one." And that's how Unix was started. And Unix didn't want to do what Multics did. It was like a counter Multics. As it turns out today's Unix has adopted many of the innovations that were included in project MAC. Well that didn't become a commercial computer. The ideas that came from that influenced the entire computer science and the art of computer science.

Fairbairn: So you were- you approached Professor Dennis who was leading Project MAC? Is that correct?

Patil: No he was in the Computer Architecture group.

Fairbairn: And so that's an area-- so tell me about how you got involved in that.

Patil: Well it was a very interesting experience for me. I went to see Professor Dennis and he said "Here are a bunch of papers. Read them and if you still like this field come back and here's a desk for you and then I will support you next year as a research assistant." And it literally happened like that.

Fairbairn: So this is continuing to pursue your interest in computers and building computers and so forth. What did you focus on for your actual Ph.D. dissertation and was that related specifically to the Multics and MAC program?

Patil: Well I should tell you a little bit before my first job as well as my first exposure to computers, okay? The first exposure was in IIT Kharagpur itself where IBM had donated a 1620 computer. And in the Computer History Museum we have restored one 1620. It literally arrived in boxes and professors and us put it together <laughs> and that's where I learned how to write Loader and assemblers and Fortran program writing and so on. So my first introduction to writing algorithms came from that computer. It was a very interesting computer in its own right. When I came to MIT of course I was a teaching assistant busy with that but then came summer and I wanted to have a summer job. And lucky for me the department had been looking for some kind of software so they can keep track of who was doing what assignment. Who was teaching what course? How much was being paid from different accounts both for professors, teaching assistants, and research assistants because there were so many of them they could never have up-to-date accurate information. So they said "Can you build a software system to do this thing?" And I looked at it and said "Sure, I can do that." <laughs> Very confident first year graduate student. I had some exposure before. So and then they gave me an account on this time sharing computer and gave me some 32 tracks of memory on the hard drive. I said "I am in heaven." And interaction was with the teletype and indeed so I built this software system for use by the department. It was an online system where you entered text and it took key words to pick up the data, collect the data, sort the data, organize data, and finally I put it out in a format that Runoff could accept that nice looking reports would come out from this. And I was so happy. The summer was long enough for me to finish the project and hand it over and it was quite an experience for me. I'd come to learn to build computers but first exposure was writing programs and I used the language M-A-D, <laughs> MAD.

Fairbairn: Is that a language created at MIT?

Patil: No, it's Michigan. I think Michigan Algorithmic Decoder or some such thing. I still have the manual with me. If the museum would like it I have it. And the low level assembly language of F-A-B for IBM 7094. It was quite a learning experience and I was on my own.

Fairbairn: Sounds like quite an achievement especially for such a short time and little experience that you had going into it.

Patil: Yeah, I was just kind of doing what seemed logical thing to do. The good thing was I followed a methodology of testing every module and created module, tested module. And when I put them together in the last two days of handing over, you just kind of put the- turn the thing on and voila, it actually worked. <laughs> It was quite a satisfying experience for me.

Fairbairn: So I suspect that had sort of confirmed some of your own understanding as far as the way to go about constructing complex systems and so forth in the future.

Patil: Indeed. And so it turns out after that I got my doctorate and went for interview. People were more interested in what I did in that summer job than all the work I did for Ph.D. I should have taken the hint, switched my field from computer architecture to software because the need was so great. Anyway.

Fairbairn: So you moved on from that. That was your first year as a graduate student. So tell me about the research that actually resulted in your Ph.D.

Patil: So we were in that Computer Structures group led by Professor Dennis. We were interested in how to build computers correct-by-construction. So at that time until that time what used to be done you build a computer, clock-based system and after doing what looked like a perfect design on paper when you implement it, it invariably required debugging involving you know Scope, and tracing things and so on. And so the academic question was, “is there a way to design so that you didn't have to do that part?” Because you had to specify correctly in order to get correct system, but you were not going to be hampered by the variation in physics or delays and so on and it was felt at that time that clockless systems could do that.

Fairbairn: I'm curious let me interrupt. "Correct-by-construction" is a term that was applied much later in VLSI design and so forth and I'm sure in other places. Is that a term you actually used at that time? Or do you look back and apply sort of later terminology to what you were trying to do?

Patil: No, that was our goal. We could easily study writings but in terms of what we were trying to do we would speak in those terms. And software people later on started using some of these terms, right? So for us it was very, very it was the primary thing, correct-by-construction. So that became sort of a way of doing- so the only way we could see being able to do that was through modules, you know, hardware modules that would connect to other modules by just interconnections without involving a clock. Now a clockless computer had already been designed and built at the University of Illinois so it's not that this thing hadn't been done, but there was no methodology, no systematic methodology, to help so we're trying to create a systematic methodology. So those modules had to be proposed and then they needed to prove that indeed anything built with those modules in the prescribed connection manner would work. If the connections were good, you know, you would not have problem with race conditions. Stuff would work. So that became the point of investigation and my assertion at that time that indeed one could come up with a set of modules with which you could build whatever you wanted and at least with respect to correctness of working in the sense of timing and the circuitry working as written on paper would be that. So I proposed modules, then I had to go prove that, and the method of proof becomes very mathematical. So indeed that is if you read my thesis it doesn't look like building of computers at all. It was really at the foundation of this. My practical work relating to it got done later as a professor.

Fairbairn: Is there anything else that you think is important to talk about relative to your sort of pregraduation before we move on to your professorial duties and other things? Are there any other events or things that you think are important in terms of later developments and learnings on your part?

Patil: Other than being in a very vibrant place where lots of things were happening including ARPANET and the Multics architecture itself and the problems they were working on, artificial intelligence was being worked on. Knowledge-based systems were being built. That all influences you so it was a very nice place, very fertile place to have been.

Fairbairn: So you finished up your Ph.D. As you were working on your Ph.D. and looking to the future, did you have a clear idea of what that future was? Did you have "I'm going to be a professor," or "I'm going to be a computer architect."? Or were you just focused on the job at hand and how did that evolve?

Patil: Yeah, you know completing a Ph.D. is an interesting process. It's like learning to fly solo. And there are times when you don't know whether you're going to see end of the tunnel and it is as much the training in kind of keeping your cool when things aren't working or what you thought would work doesn't work. It's as much a human development as it is the advancement of science. So now I didn't have much time to think what I would do afterwards. I thought definitely I would do something. A very interesting thing, while I was doing my Ph.D. Larry Seligman was my office-mate. One day Larry says "Suhas, what are you doing here? We're starting a company where we're going to build computers. Do you want to build computers? That's what you do- why don't you quit and come join us?" Well he and the five of them then built Data General. <laughs> So at that time I said "Look, I came here to finish my studies. I mean this sounds very attractive but I've got to finish my Ph.D." It would have been just fine if I had said "Okay, let's go build a computer." <laughs> So that happened. But anyway I was actually teaching already not as just teaching assistant but I had been appointed an instructor. That was very special because very few, like two or three, in the whole department of teaching assistants are allowed to become an instructor or lecturer. And so you actually join faculty. You are a junior faculty. So I was actually teaching in bigger classes or taking subjects on my own so that was- it's one thing to be TA, another thing to be responsible for a class. So I had that kind of experience for a couple of years and definitely the teaching was an option. I didn't know whether anybody would offer me a faculty position. It was not known, but I definitely wanted to explore that and the other option was to just go to the industry. At that time I was firmly thinking that I would spend some time in industry and go back and maybe build a computer in India. Because my thoughts still I had not said "I am coming to America to settle down." And as it turns out I was so pleased that Carnegie Mellon and MIT. Both of them they gave me an offer to join the faculty which I thought wow, I mean can you imagine? First I get an admission, I go to study at MIT, I get my Doctor of Science, I don't have Ph.D., I'm Doctor of Science, and then I'm invited to join the faculty. It was just kind of unthinkable. I was just pleased so I don't know how I decided to remain at MIT and still I don't know but anyway.

Fairbairn: So at that time when you made application to CMU and to MIT did you also sort of actively pursue industrial options?

Patil: Yes I did.

Fairbairn: And so among those various options you decided that MIT was the- however you decided.

Patil: So what happened, you know Bell Labs had been tracking me and they do a very good job. They had a consistent person come and see me and meet other students, so I had a connection. So as I was about to graduate they said "Well come into Bell Labs and I went to Bell Labs. I also went to Indian Hills. They were working on CTS5 at that time. So I was expecting to receive an offer from them and that was more likely to have in my mind it was more likely place I would go. And as it would happen things change always in departments and companies. They call me up, you know, I had an offer working for you but the department is reorganized and we're back to square one. Yeah, I'll have things for you but it's going to take a little bit of time, okay? While this is happening, I get an offer from both schools and I said there's no looking here or there, it's going to be one of those two. I just then accepted and became a professor.

Fairbairn: So you mentioned making a decision to stay. Did you ever decide to stay or did you just end up staying in America? At some point, you decided this is where I'm going to make my career or did that just evolve and become inevitable?

Patil: No. At one point, I decided. That's also very interesting historical point from computer history's point of view. So here it is, I am engaged in ways of creating and building computers, which I'm sure we'll talk about later, and I'm thinking, well, I've got kids and, before they get a bit older, I should figure out what I shall do in India. ??? Tata was the head of Tata. They called me up and said, "I'd like to come visit you" so he visits MIT and so they said, "Why don't you come down and talk to us and others." So I did go to India and interviewed for a couple of companies. One of the companies was DCM in Delhi. So I'm interviewing here. I don't quite remember the name of the person. They're trying to build a computer. They asked, "Did you know how digital computers work?" I said, "Sure." Okay. He started asking me questions about they wanted to build a copy of Digital's computer. I said, "Why you copy? I'm an architect," okay? "Let's decide to build a computer and we'll build the next generation of computer right here. No need to copy. We can learn from what the field has done and we just do our thing." He just couldn't even fathom to build a computer that would advance the state of the science. When I understood all he was trying to do was to see whether I knew enough to copy, I said, "I'm not interested in this at all." Then, after looking at things in India, it turned out the political situation was not very good. Elsewhere I went to interview, this is another interesting experience, they said, "Well, you left us, you went to America, had good time there and now you're going to come and you're going to be our boss?" I said, "Hm, isn't that a message?" If I were a football team, I would like to have best lineup of the world anywhere. I was already thinking American. My perspective has broadened, my friends are in America, I actually belong in America. So I'm not going to go back to India to make a show. Here it is. I will just...

Fairbairn: When was that?

Patil: This was in 1974/'75.

Fairbairn: At that point, you were just completing your Ph.D.?

Patil: No, I had finished my doctorate in 1970.

Fairbairn: And then you had been a professor at MIT for several years?

Patil: In 1975, I went to University of Utah from MIT. So it was that transition time.

Fairbairn: Tell me about the five years you spent as a professor and what you focused on. What were your interests? How did your interests develop during that period? What were the new things you found exciting at that point?

Patil: So I had done theoretical work in my doctoral program and my interest was in finding ways in which can build computer- build it correct by construction and my hope was 100%. Also, I realized that technology, the way it was progressing, the integrated circuit, which already was there, would be the mechanism that you would build a computer with. In fact, I, at that time, could see that whole computers would be built. So, unless my method applied to the building of a computer as a chip, they would just remain theoretical studies. I started translating the theoretical ideas to structures, circuits that would lay out in two dimensions because that's what integrated circuits were. They were all surface. They were not three dimensional. The transistors were three dimensional but the fundamental circuit was flat. That translation is what I worked on and successfully found array structures where one could implement whatever digital circuitry you had desire of building. This was then very regularized PLA-like, Programmable Logic Array-like, okay? Programmable Logic Arrays are like columns and rows. To build a complete system, you would use them as modules and interconnect them. My idea was one uniform thing where you would be able to do everything, to build the whole thing. We had to introduce notion of memory, like flip flops and other memory elements and be able to do that in a flat surface. That work, conceptualization, formulation, and building it out of discrete components to be sure the ideas work, and this was all without using clocks, okay? So I actually completed that work at MIT. I was trying to build this thing on an integrated circuit but I found, even though MIT had such facilities in Lincoln Lab, I really wasn't getting anywhere. If I had a DOD project, maybe I could have done but this was, by that time, National Science Foundation work. An opportunity came up in Utah where General Instrument had donated a FAB and the university was interested in creating VLSI program. I said okay, you know? Yes, it involves going to the west and going to Utah but I knew pretty much all the faculty members there professionally...

Fairbairn: So before we get to your transition to Utah, you had done your PhD work on very theoretical sort of computer architecture description. Now you're talking about actually building things in circuits so there's a huge gap there, if you will, between those two areas. Was there anybody else you were working with that shared your interest in LSI, VLSI implementation? As a reference, 1971 was when the first microprocessor, the 4004, was introduced. The idea of computers on a chip was new to most of the industry. Were you the pioneer in that regard in your group?

Patil: I was the pioneer. In MIT, the department was electrical engineering. The folks who were in the computer section, project MAC, Professor Jack Dennis was the closest to digital circuits. There were other people in electronics in other sections but none there had the breadth of knowledge as I had because I'd studied electronics, including communication, radio, valves, transistors, circuit structures. I'd done significant programming, okay? Then I had fairly good knowledge of circuits and even devices, if I had wanted to kind of go into material science, right? This is where good grounding at undergraduate level is very helpful followed by very good grounding in graduate school and not having a very narrow interest but broader interest, right? I was always a builder. I built, in high school, projects in my undergraduate. I just didn't build any physical things other than the program I wrote in my graduate education. For me, it wasn't like I'm doing anything so very special. We would like to do this thing and systematically go about it and I had the lab and my group at MIT. So it turned out that was a transition time for the computer science guys to get involved in design automation as well as physical design. So, to that extent, I was leading, so much leading that people had difficulty, other than my colleagues, understanding why I was doing those things. But, as time would tell, in fact, the world was going to go that way and I was actually working on the right things.

Fairbairn: So you were headed in that direction and Utah ended up being a place where you could pursue that. How did you make those contacts? How did you come to know the professors at University of Utah? How did that transition happen?

Patil: So, as a professor at MIT, after I built these solutions for clock-less circuits that were flat, I actually physically built, okay? It was something that I could put in a suitcase and travel. So I actually traveled very extensively because I used to get invited by the universities and companies. I traveled in Europe, I traveled in America, I had given a talk at the University of Utah. Also, from a historical point of view, interestingly enough, at my graduate school time, much of the computer science activity was sponsored by Department of Defense. ARPA, it used to be called ARPA then, DARPA today, they were very forward-looking. The history of computers in America, if you do not see what kind of sponsorship and input they gave as well as freedom they gave the computer science people, you don't quite understand how this country became so strong in computer science. They had another interesting idea. I don't know whose it was, whether it was Licklider or whether Kahn’s, you know? They said, "Let's take the graduate students, let the institutions pick the people leading the field and get them together." In that context of this collegiate relationship because DARPA was supporting Utah's graphics work was going, other kind of things, so they brought these graduate students pretty much in the last year of their studies to one place. Well, if you go to the list of those who were present there, you'd be amazed. They basically set the history, you know? They led. It included Alan Kay in there and so on. We thus had friends throughout the country. Then there was a conference done where some of my ideas were presented in Catalina Island. A lot of professors attended. In some ways, the size of the computer science community was not very large in those days and it was possible to know those who were working in this country. Through that means and by visiting and giving talks, I had met most of the faculty at Utah somewhere else. They were familiar and they said, "Why don't you come down, give a talk" and, at the end of the talk, they said, "Would you consider coming here?" I became aware of the lab and then I gave it thought and talked with people. They said, "Come in, you can come here. Instead of staying in an apartment, you can actually buy a house and raise a family. Besides that, we have this fab and please join the other guys for the VLSI program." I found that very attractive and so I chose.

Fairbairn: So, in 1975 approximately, you then moved your family to Salt Lake City and...

Patil: Got on a U-Haul because the planes had gone on strike. Packed up my small belongings, whatever it was, my family in a truck and I drove the truck across the country ahead of the snow storms. Today, you say, "Why would you do that?" Exactly. I arrived on Christmas Day in Salt Lake City.

Fairbairn: Oh, my. So part of what you mentioned was there was a Fab there that had been donated?

Patil: Mm hm.

Fairbairn: Tell me about the evolution of your work at University of Utah. What directions did it take and what did you focus on?

Patil: Yeah. In the fab, you know, there were earlier General Instrument employees who were totally in the field of devices and making fabrication, in particular, you know, Ken Smith, who then also joined me as a professor in the department. He had these circuits and the low level circuits and there was one more professor who had the device level. We set out to create a program, got National Science Foundation support for this work, to translate what looked like, on paper, working ideas to actual implementation to see whether those ideas worked because there was a lot of technical risk. People said, "Well, sometimes stuff works or not. This whole array business, which was very much a departure from others. I need to now explain this whole thrust, it was clear to me and also other people, Carver Mead was on the west coast who was interested. As the complexity of the chip increased, as the transistor counts, because, in order to be able to cope with the design and not have it become, you know, so large that it is not able to produce, needed the chips in a timely manner, so it was all handling complexity, the introduction of methodology of design, you know, the design automation, use of design automation. So now my structures, circuit structures come together with oh, okay, let's have methodology of design, let's have tools that help but let's first make sure the structures work, okay? This is really where we took and translated the ideas which were implemented as a proof using digital components were now translated into a working prototype on a chip. We used at that time i-squared technology, which, of course, is not in use today but that was available to us and we used that. That gave me, as well as all my colleagues, solid proof that these ideas would actually work. At that time, we hadn't built any design automation. We were just doing everything manually. My approach to this thing from the beginning came from as a computer architect, as opposed to circuit design. Carver Mead had started from the material side as well as circuits and he was going up to higher levels. I was coming from computer architecture, language, way of expressing. So my approach was to sort of find a level of abstraction where the system designer, the architect, could express what kind of system had to be designed or wanted to be implemented. With enough information so that a machine translator, you know, like a compiler, could translate that into finished GDS tapes for it to be made. So these two schools of thought or the schools of approach, mine coming from the top down and Carver's going from the details to abstraction, were being worked on at the same time. This came under the terminology of “Silicon

Compiler". The methodology Silicon Compiler, the development of high level languages like VHDL and later Verilog came. These were all interrelated things. The lab in Utah helped me build the first prototype then we were given a challenge of trying it out on a big piece, so to say, bigger problem to iron out any other issues that might exist and get a better hold on the higher level language and the translation process. That opportunity was given back in General Instrument who now had become a sponsor in addition to National Science Foundation. They extended their facilities, including their main fabs, for this purpose. That turned out to be a very interesting project. They had a company called Jerold, a division, and they wanted to try out set-top boxes. There was already a contract out to a regular semiconductor company to go build that thing so there was no risk if we failed. We were given the same pieces of data and they said, "Here it is, real data, now try to build something and this will give you real feedback as well as a chance to advance your science."

Fairbairn: So when you say given to you, this is you and your graduate students...

Patil: Yeah, at the University as a project.

Fairbairn: ...who had been working on this design methodology which included tools based around the structured logic array that you had created as part of your research?

Patil: Yeah. By that time, this was called Storage Logic.

Fairbairn: Storage Logic. Right.

Patil: Because it was sponsored, I was looking for real life problems so they gave me and my students and my colleagues all worked on this thing to build our "real first chip file", my own experience. As it so happens, even though this was a novel way of doing things, we actually finished the job, even while the mainline part was still working on finishing the layouts. We actually had our part working and, since this was a trial, the company chose to say, "It's a trial, let's go use this thing to do a field trial," which, of course, pleased all my colleagues, myself and students. 20,000 units. I mean, for me, 20,000 units is just like my goodness, okay? But 20,000 units were made and they were actually deployed in the field to get feedback before them developing their final version of set top box.

Fairbairn: So, before we pursue that path, I wanted to understand one other thing about this chip. You had said earlier you had spent a lot of time and effort on clock-less designs and trying to understand how to build those. Was this an example of that or was this a clock design using otherwise the methodologies and so forth that you had developed?

Patil: So I'm trying to remember. I think it had both clocked and clock-less circuitry because one of the things was there was a microprocessor that was in existence they were using so we had to actually implement that. We implemented phase lock loop for tuning, we implemented this microprocessor and so on. We actually ended up doing a mixture. As it turns out, in subsequent chip designs which I did for another decade and a half whenever I did designs, there were sections that were clocked and there were other sections that were self-timed, particularly 5/4 kind of things and the communications, in the closet of the communication with clock-less. In a structured data part, it tended to be clocked.

Fairbairn: Okay. So you had created this "test" chip or prototype chip that all of a sudden they wanted 20,000 units of, which I guess they built and used successfully in their trials. What was the next step? Lead us through the path from university professor and graduate students doing work for one of your sponsors to actually going out and starting your own company.

Patil: So when the chip was successful, then, obviously, the question was how do we make it usable in the industry? General Instruments said, "Look, we'd like to build systems on a chip but, in order to practice this, there needs to be design automation tools and it needs to be fleshed out fully." University work was kind of done so I tried to get industry to adopt it. I visited many companies in Silicon Valley, leading companies. Silicon Valley was much smaller then in '79. I would get great response from engineers but they would say, "Well, but I could not convince the V.P. of engineering because there is too much departure from the way we do things." So I was not able to successfully hand it over to somebody, even though I promised I'd give you students and I would consult for you because it was just too different a way of doing things. General Instruments, which had actually seen and used, they said, "Well, you know, you cannot do too much more at the university so one of two things have to happen. Either you come, take a sabbatical, come and join us or leave university and join us and build the whole design automation around it so that we can make systems on a chip easily or, if you have any other ideas, let's think about it." Since I was in close communication, I said, "Well, I cannot leave my town because of family reasons but I would really look forward to building this thing outside university in Research Park if I could have funding or a contract." That was appealing to them and they gave me a contract. I tried to get a leave of absence from the university, which I cannot understand why they turned me down, so I am faced now with a real hard dilemma. Do this thing or quit. So I finally came to the conclusion, well, this thing had to be done. I mean, I wanted to see it to fruition and I said, "Look, I have to resign." I maintained my affiliation but I was no longer on the faculty. I moved to Research Park there in a couple of buildings next to Evans and Southerland with the funding provided by General Instrument. I set out to go now build the entire design automation system and the whole methodology. We were given two years to complete that task and do real chips, which we did towards the end of that.

Fairbairn: So what were the real chips? What became the next project to prove out the methodology and so forth?

Patil: So literally, about two years, we were doing the design automation from the ground up. Unlike other design automation work that was going on at that time, which was really Calma, drawing things were there, there was place and route work was going on and so on. We were doing silicon compiler kind of work but we had to build the entire system, starting from the graphics to the language to simulation to the compiler and so on because the methodology was different. The only thing we could use was bulk list tools for doing design of the cells, which we did. Unix was used for our computer operating system. We did that. The opportunity came up to do the next generation of set top box chip. Before, they had a whole board and they wanted to reduce it to literally a tuner and a single chip. This was now not for jokes or trials, this was, "Can you do it?" "What are different ways of doing it?" That job was won by us. We were engaged to do that for General Instruments Corporation. This was to go into what became later on known as Starcom V in a set top box, very highly successful. This was the first real chip, other than trial stuff we did in between, to use the tools all the way from the simulation to the entering to the compiling and creating GDS and going to fab and so on. That chip was done in 1983.

Fairbairn: Was this a purely digital chip? Did this include both analog and digital?

Patil: It had some analog but so very little. It had to deal with the tuner outside. We had to interface to it. It was a crude digital to analog converter in there. Otherwise, it was all digital.

Fairbairn: So you spend nearly two years developing the tools and designing the chip? Or subsequently designing the chip...

Patil: Oh, we did trial design during that but then, the third year, we did this chip.

Fairbairn: Okay.

Patil: Tools never stopped from being continuously improved, especially you get the feedback from doing the design so you can feed that back. This complex chip was designed, literally, starting from Labor Day and the design was finished around Thanksgiving time, which, in those days, it was considered a very rapid design. The fabrication was done in December. We had the parts in January. During this time, we had concluded all the VCs were in Silicon Valley or Austin, Texas. I would try to make an appointment when they were coming to ski but there was no good engagement. We figured out that this activity, in order to get support, we really had to move to Silicon Valley. So we used that...

Fairbairn: So, by that time, you had come to the conclusion that you really wanted to grow this company and General Instruments was merely a path to a much larger future? Is that the vision you had?

Patil: Well, that was always on our mind, including from the very beginning. When we started, the arrangement was we would build a tool. What company were we building? We were building a company that will use this tool to go build chips. General Instruments' Fab, run by Ed Sap, in Hicksville, is that, "I'd be happy to sell you the wafers at fixed price." Now, you're talking about this is 1980. 1979, when it was conceptualized, and the company was started in January 1980. At that time, the company was called Patil Systems. I had a wafer source at a fixed price. They said, "If it meets parameters, you owe me money." I was very confident we could go make chips and the PCs were just coming, very soon PCs would come. They said, okay, there will be a lot of need. So it was conceptualized as people who will rapidly do designs for this. Now, there was a component of that, the needed thing was the design automation. A lot of people were not clear, are we design automation company or what?

Fairbairn: So just to put this in context and make sure we understand, there are a number of things which you have just spoken about that were revolutionary concepts at the time. I mean, even talking about systems on a chip in that time frame was not a commonly accepted thought. It sounded like you were working with a company that had an advanced vision of where semiconductor technology was really going and what impact it would have on their electronic product. At the same time, you were talking about doing design tools as an aid to really building a chip company but you never had any plans to build a Fab, you were going to use somebody else's fabrication capability. Some of the things which only many years later became commonly accepted ideas. Talk about that. Did you realize at the time as you must have what a change in thinking that represented? What kind of risks and so forth did you think were entailed in that?

Patil: Yeah. That's a very good point. Sitting here, that was definitely a historically important period, Silicon compilers being worked on, system on a chip. To talk about system on a chip was around in the field. It was not like I woke up one day and said, "We should have a system on chip" but very interestingly, there was executive VP in General Instrument, Larry Hill. He was an amazing person, technically knowledgeable, knew how to engage academic people and work with smaller companies around him, which, many times, he sort of said, "Why don't you start a company? We'll work with you." He attended to the business and technical needs, particularly technical needs to support business. This is a billion dollar company, General Instruments, their business, right? So he had led effort in the company to persuade the company that the set top business, because they were very price competitive, you know, the set top must not cost very much. The only way to achieve that was to do system on a chip. He had these labs, they were all under him. He found, in me, ideas that he could relate to. That's why he was my mentor, sponsor and almost godfather in General Instrument. This was being done from the headquarters of General Instrument. He had a need, I had technology and one good thing about professors is you do tend to study. You study what's going on in the industry, where things are, where technology might be going. This is how you figure out what to work on. I had been doing that as my normal work at the university, tracking technology and so on. These ideas were coming and I said, "You know, today, the tools are not there to go build things but the Fabs cost a lot." I was just trying to look for an angle if I'm going to have a company, I should have some way I can do and I'm not in Silicon Valley. So I said, "You know what? Somebody's willing to send me wafers, at a fixed price, which is very important." In order to build a company which relies on other people's giving wafers, their needs could go up and down and I may have that source and I may not have the source. So my design ought to be capable of being made in more than one place and I should be able to migrate the design using computer means and not having to reengineer, literally push of a button and do an overnight run, a recompile and you should be able to target. That way, I'm not going to be stuck. The test program, no matter where the chip is made, should be exactly the same which means your tightness of the end result should be within the guard bands so you actually have chips that yield, right? These thoughts of what it takes to have a fabless company, I think I was the leading authority on thinking, you know? I would talk about it but it became that, many times, taking ideas from university and bringing it to industry requires the professor or students around him to actually do the transfer by going to industry and doing it because the gap between what was being practiced and what was being proposed was too much to be bridged easily. If I had come up with a better routing algorithm that did 10% better than the industry, it would have been accepted very easily but to do it entirely different suggest that the chips should be designed by the architects and the system engineers and the role of the circuit designer was in creating technology in the form of cells and everything would happen using computers. Then you would be good enough in order to be able to survive as a maker of semiconductor company using other people's, for some reason, why would anybody give you those wafers to begin with? So I had found out that our needs would be small, that it would come just in the noise. The reason people would consider giving me wafers was because of curiosity, this new way of doing things, maybe it will be important to us. So even the Japanese were willing to do that and General Instrument had a more direct interest. They had supported my program. They had this division that wanted things. The semiconductor division were very happy for somebody else to go do the designs and the fabbing be done there. So I had a small ecosystem within the General Instrument but I found that even there was acceptivity[sic] beyond. I was not 100% sure but it looked like that. It turned out to be true afterwards.

Fairbairn: So the people who were in the fab, Ed Sack, you mentioned, liked the model of just being able to sell wafers at a fixed price where he didn't have to worry about yield and other things, is that correct?

Patil: Correct, and because it didn't amount to very much of their...

Fairbairn: Of his total production.

Patil: Production. And it reduced the complexity for him as a sister division of General Instrument. But he told me clearly, "It is not just for General Instrument wise. If you make a business of it I'm happy to do business supplying you wafers." So from that point of view General Instrument was the first company as provider of fab services. They did it for me.

Fairbairn: It sounded like they had a number of innovative thinkers in important places.

Patil: Yeah, they were leaders at that time.

Fairbairn: So you went off and you designed these chips and it became clear that you really had a company here now. You got to the point where you got the chips back in January. Complete that story. You were talking to VCs, couldn't get them interested; decided to move the company to Silicon Valley. Tell me about all of that.

Patil: Yeah, so what happened is Atari was a big customer of General Instrument and there was a slowdown in industry in '72, '73. It affected General Instrument so they had to cut back. General Instrument's notion was to keep giving me more and more jobs in the contract. We would not just do just a few things. I could do other things, but at least that was a very helpful thing for them, and I was very happy to do that. Then they said, "Okay, we're giving you additional contract, but go look for money because we're not..."

Fairbairn: This is '82, '83.

Patil: Yeah. It was in '83. They said, "Maybe we will participate if you get a VC. That will be fine because we're not VCs." We know that as company people. "But you do financing we'll participate in that, so go out raise money." As it turned out the venture capitalists were in Silicon Valley, or as Boston Silicon Valley or Austin, and the notion that was one of the leading efforts in semiconductor might be happening in Salt Lake City kind of didn't match Silicon Valley. So they went, "Who the hell are you? You're too revolutionary." Both meant trouble. It means that I would not get funding, right? So we felt that in Utah where we got money from angels, both from Silicon Valley as well as all over. My mentors gave me money, family gave me money, employees, father, uncle, I mean, this is how we bridged our gap and so we were basically now struggling at the end of 1983. So we sat down. We had finished this design, and I gathered my team which wasn't very big at all. So we were all together, seven people. I said, "Guys, unless we do something we're not going to survive. So I don't know how many of you want to come with me to Silicon Valley and your situations. Let's do a transition. So let's start preparing for the transition." And we used the time when the chip went for fabbing to actually pack up, literally, call a moving truck; move our stuff to Silicon Valley. We had a sales office here that became our headquarters, and we made a move.

Fairbairn: So how many people actually moved with you of those seven?

Patil: So we were in the end four; me and three other guys who are now all recognized as founders. And then what happened the other folks two of them said we'll go do our masters, by that time you'll be established and we'll join you at that time. And one person says, "Well, I've got to take care of family, so I've got to get a job." It turns out all of them, once we get established, all of them joined us. It was just phased joining. So interestingly enough we came here. Now this is an interesting story. So it went into fab and when it came back-- this is just a little hilarious story I want to share with you. So the chip comes back. We're all excited. I had built a little tester which was all made of switches and LED lights, and I put it there, and I'm first trying to test the communications side of the chip, and that chip is dead on arrival. And I said, "Oh, my goodness," and we have such a morose attitude. I said, "This means trouble." And interestingly they had not sealed the cap.

Fairbairn: The top of it, yeah.

Patil: So we had some of them with open, so we opened it and noticed that the back-bias generator didn't seem to be in the right place because even with our naked eyes we could tell something was not right, and on further examination realized that the die had been mounted 180 degrees out of phase. And because the bonding was different it just had the wrong things bonded. So we couldn't just turn the chip around. So we said, "Oh, maybe we have hope." We ran to a nearby company on Bordeaux Drive and that company had a scraping tool and a bonding tool. So we re-bonded it correctly to the scheme we had, and then we turned the chip 180 degrees. It was a DIP-switch so it could be done. We plugged it in, started testing and my goodness the stuff seemed to work. And so after doing a certain amount of testing, because we didn't have any automated tester, we said, "Okay, let's just put it in the system. The system was in New Jersey, so my engineering person got on a plane, took it there, put it in the system and it only took three days to bring up the whole system including software. It was magic to us. So that was at a very crucial junction in time for us. We had a proof of a working chip, so that this otherwise revolutionary thing looked more practical, more doable. And the model was still under question by many people, but then Fred Nazem who was a VC from New York who also used to come here, but his background was different, it was in Caltech as a nuclear scientist. He looked at it and looked at me and my background and he says, "Maybe the professor is right after all. Maybe there's something in it," and then he started helping me. So that's how my engagement with the VCs happened. Then slowly in addition to my own students now other people started joining to build a real company here in Silicon Valley.

Fairbairn: And in the process you changed the name, and Mike Hackworth joined you.

Patil: Yes, Mike Hackworth. Now with some VC connections I could go meet people who had a substantial background who could be mentors. So Mike started out as being a mentor to us, and the more he looked at it he had those thoughts. He was interested in design automation and he might have been just thinking on his own in a fabless because he had built fabs and he knew fabs, the difficulties, financial difficulties with fabs. So as he helped us in the business side-- and see his background is marketing, the operations background, and sales background, and coming from technology. There was one time at night when he was helping us on the business aspect I said to Mike, "Mike, if I build can you sell?" He says, "No problem," and he turned around and says, "So you think you can build?" So if we can do both we have a basis for taking this thing forward. And six months later he joined us full time.

Fairbairn: So there's some other things I want to get to and I think those are the important beginnings of Cirrus Logic. Are there other milestones within the company that you think are important sort of to sort of complete the picture in terms of either the technology evolution or your own personal evolution?

Patil: So yeah, let me share with you what people don't realize. So we said fabless semiconductor company. That idea in evolution of semiconductor industry is very natural, but we couldn't see it that way, and I had an idea so I was pursuing. There are phases in industry. Whatever is the most difficult thing to do that is what defines what you have to be good at in order to do. And what becomes a very easily practicable art then becomes widespread and it becomes a feeder or supplier to it. So it was unthinkable before that if you did not have a fab, to build a semiconductor company without fab because they used to run very specialized processes. Now imagine CMOS was coming at that time and digital stuff was becoming 100,000 transistors. So the problem why others could not see building fabless semiconductor company and could not relate to it was some very, very key factors which I'd like to kind of share with you for others to see. Firstly, I already mentioned that if you did not have portability with machine automation the time and effort required would put such a heavy burden a fabless semiconductor company would be in trouble. Second one, it was very vital that as the complexity increased that the system architects, and the system knowledgeable people being able to completely define without involving, without getting circuit and throwing it over the while to the layout people. That they need to be able to play around, and translate, and optimize, and do the correct architecture of it themselves at a higher level so that they could, in fact, be the designer. And that would make it possible to do the bigger designs in equal time, which was essential for a fabless semiconductor company because you were going to rely on IP (Intellectual Property) more. So unlike process capability we're now shifting to product IP, design IP, the speed of design; the flexibility of being able to make your products elsewhere. So since I had been engaged in design automation, methodology of design, had actually built chips by now, so I could see what all elements were necessary in order to successfully carryout. And fortunately PCs were becoming popular and the early PCs had too many components. So it's clear that there was need. The processor was there, of course, but all the IO had to be strung into their own modules. So there was a good sense of the need in industry and a growing market. So this thing could be better served by people who were experts in graphics, or communications, or disc drive systems people. So I could see in my mind the equation working, and I had Hackworth who had joined who was a veteran in the industry, and from his knowledge and perspective he could see. And he could see how somebody would pay for it and not ask us to price it based on the square millimeter of the wafer used, so that we had that whole element in place. The design automation industry was not advanced enough, so I actually had, at that time when I showed up in the Valley, the best design automation system that would do from A to Z, cradle to grave. Now, if I had been positioned differently and I had different kind of backing we could have gone to become a design automation company, but you had to understand. You cannot sell religion. My methodology was almost like you had to buy into it, so if it was an extension of existing methodology improvement maybe I could build a design automation company, but here we were asking people to change their ways. I don't think it was possible. So the only option left for me to take this forward was even though the rest of Silicon Valley didn't believe, honestly, Silicon Valley thought we made great proto-chips, but we would not be able to build competing chips, particularly when volume was required. And while I'm telling them, "Look, set-top business is extremely competitive business. I've done it. And we rechecked our method. It has proved; we were brutally honest to ourselves. We had other people look at it and we couldn't find any flaw." So we said, "Well, you've got to believe in your own assessment," and we took the risk, and we found quality VCs to back us up. And so that is how we proceeded. And we even had this comment by Jerry Sanders once who said, "Real men have fab." It was directed to Mike and me, because in his own experience, in his mind you needed the help of tweaking at the fab in order to bring the yields, and we were saying just the opposite. Remember, I'm coming from a computer science. I'm saying, "No change in process allowed," because somebody's doing me a favor sending me a boatload of wafers. He's not making too much money. Later on, of course, TSMC becomes very successful. So I did not have control and yet I must design competent things. And people say, "Well, how can it be cheap?" Well, this was too sophisticated an argument. The argument was system designers will design in such a way as to do more optimal design than what it was capable of doing before. They would find ways of doing things faster. There would be fewer transistors actually needed to do the same design, and then because there was a total control over design rule used because that had to be done only at a cell level which were very few building blocks. Our total control over adherence to the rules that are agreed to between the fab and the circuit engineers could be far more adhered to because then our debugging would be reduced, and our chances of failures would be reduced. So all of this in my mind and Mike Hackworth's and other's minds it translated into, "Yes, we can deal with high volume applications." So contrary to the belief of the industry, because this was too much a change for the industry, guess which businesses we won first. First was set-top business, second was disc drive business. Can there be anything more grueling than that? First chip, that set-top box, six million units were made in the mid '80s. That was a pretty impressive number.

Fairbairn: That's a huge volume in that timeframe.

Patil: Then our disc drive chip, first single die, 20 million units. I think it was in a span of 18 months. So obviously our assessment was correct. By the time we do this thing now TSMC comes along and others and so your supply becomes widespread. I used in Cirrus Logic General Instrument's fab, Atari's fab, almost all Japanese fabs, fabs in Korea, fabs everywhere, and we had to because it's the only way we could meet our growing needs until TSMC came. Do you believe we used to take 20 percent output of TSMC? This was a fabulous thing. So anyway, so this is how it worked. The reason it worked is that we had the advantage of tools and methodology. Later on design automation came with other methods and Synopsys came with its translator and then Verilog came into being. So that model was already there. The tools became available so that this model could be practiced by...

Fairbairn: A wider set of companies.

Patil: Broadly, broadly.

Fairbairn: Right. So I would like to sort of move on. Are there any other developments that within Cirrus Logic before we move on to some of the other sort of follow on activities that sort of complete the Suhas Patil picture, which I'd like to move on?

Patil: Now Cirrus Logic became fairly successful. And in the mid '90s we had a problem on hand. Even when we made one product with one name, like a disc drive controller chip, in reality it would be either start out with one product it would become two, three, finally a dozen small variations over the same chip because the customers say, "Oh, everything is great, but could you just add this thing?" Now, as soon as you add even a small amount of logic your die is now a different die, and you have different inventory, different production and so on. And in a semiconductor company the problem is you make money, but if you're left with inventory and do inventory all your profits go in the tank. We were doing a hundred masks a year at that time. As the mask costs increase the geometry shrinks. It's just going to be untenable. So I said, "Look, we've got to find a way in which the same die can meet the needs of the industry much more," so fewer dies and more end products. Now you had FPGA which is totally flexible, but then don't have the...

Fairbairn: Cost effectiveness and density.

Patil: Yeah, for mass use such as disc drive or graphics, or this communication stuff, and take too much power. So with my background in parallel processing and the computer background I was making a technology assessment. I said, "Look, how we do designs? We certainly I can see now we're in the range where we could build such a multi-core DSP communications processor chip which can do same kind of performance as I would now design using gate level logic, register level logic, but I do that entirely using executable code. So this kind of insight comes to me in mid '90s, and I assemble a team. I said, "Let's go explore this and let's see," so we would need this thing in five years or so. So the multi-core--and this suggestion that you would do a high performance chip as an alternative to this traditional methodology which I was practicing what I had kind of developed myself. But now as an executable parallel code and be able to program it, it was just too much of a leap. But I now, instead of being the professor, now I'm in Cirrus Logic and Kenyon Mei the engineering VP and myself we started supporting this skunk activity investigation and so on to go build multi-core. Nobody believed the need for multi-core or the capability of doing this thing. So that effort like two years of R&D was done on this thing, and then later on it was spun off as Cradle Technologies to do that. All the research part was done. The proof of concept was done outside the company, and the real chips were done. Now, today multi-core is a very established concept, but mid-'90s or late '90s it was not there. And it has been proposed not just as a little faster DSP, but as a total alternative for making the pictures. So these things work for a large computation. At the same time the use of the chip has become computationally more intense. Everything has been done using algorithms and so on. So shift in what the chip is supposed to do, not a controller, but a computing engine and the multi-core as, you know, we can stamp out things, but to make them work harmoniously to write programs that can use fewer modules now and more cores later all of that required design automation again. So that is how the industry I envisioned and it started to work, and the industry on its own afterwards as they met with the problem they started going there. They have not understood the full need and the power and the capability of multi-core as an honest architecture for standard chip of high performance. I think that's really what we fully flushed out and become an industry norm in the coming I don't know how many years, and beyond that I don't know. Because, see, the geometries are becoming very small. Maybe we'll have to come back to clockless designs because the delays are now just too difficult to manage. CMOS with multi-layer of metal gave a lot of forgiveness. Before we used to use polysilicon to make connections, so then the delays were varied quite a bit. The multi-layers of metal with the CMOS gave us relief. But now the geometries are getting so small that even we're not able to control that well. So maybe in certain critical sections we'll have to resort to clockless thing. And the multi-core fits in very well because you design one thing and repeat it several times and you do this in the arbitration section, or the interface sections you do clockless design. And this is how I think that the far more regularized architecture where you change the mask a few times, but a whole bunch of products come out of it. That's how I see on a more immediate next five to ten years industry.

Fairbairn: Right. That certainly seems to be the major area of new development, and still we have software problems and other things associated with that as well as the design methodology problems. Sort of working towards conclusion here I want to explore some other aspects of your career and your contribution to not only in the technical area, but also in the complimentary people humanity cultural area, and one of the areas is the creation of this organization which is called TiE. Tell me, what does that mean? How did the ideas come about? You're one of the founders. Tell me how that all happened.

Patil: So here it is, you know...

Fairbairn: First, what does TiE stand for?

Patil: Okay, TiE stands for The Indus Entrepreneurs. It's a not for profit organization and it is a way for those of us who have succeeded, not just in Valley now, throughout the world, to give back to the communities, society, particularly those who aspire to build companies and create prosperity in the community. We were successful ourselves. There were others who were successful, and we wanted to do something. Now, here it is. We are in Silicon Valley and we are from the Indus Region which is South Asia, India, Pakistan, Bangladesh and Sri Lanka. So we said, "Look, we got education. We established ourselves. We've succeeded. How can we give back?" And this giving we need to think through, so we actually met once a month for almost 12 months to sort out our thing of what could be the most effective way of giving.

Fairbairn: How many other people were involved in these discussions?

Patil: There were about 20 people. We deliberated and because of our success we had the capacity to think through it and not rush into it, and we came to realize we do philanthropic work, but to be able to pass on knowledge learned through the school of hard knocks on building enterprises, startup companies and saying, "Okay, how can we give it to the generations?" So it became to inspire the next generation of entrepreneurs, to provide them with the mentoring that is needed. It's a situation where the successful and the aspiring can meet. And it's very difficult to do this alone, but collectively we could do a much more effective job. And then we said that we set certain values which actually reflect a combination of our values coming from the Indus Region as well as our values as Americans. And in that sense we said it is going to be an open, yes it is our giving to first Silicon Valley, later on we'll go everywhere. It’s and open and inclusive organization. It's not like it's meant for the South Asians. It's anybody who wants to come. We don't even ask where you belong to. And we had two kind of members, the ones who have succeeded who were in position to give and a willingness to give, willingness to mentor. They are the charter members. They're the givers, and those who want to learn and be inspired and use the network. Networking is very powerful. As we debated this thing it became very clear that someone had to take a lead, and the group basically said, "Suhas, are you ready, or you don't have a choice." And so I then started giving it shape, working with the people organized and I was the founding president. And all of my experience in building Cirrus Logic was all utilized in how to do it, how to set values, but it was not one man's effort. It was a whole team. And today it became very successful in Silicon Valley. Also the times were right. The '90s, it started in '92. We had deliberation '92, were activities, a first time meeting called TiEcon '94, and then there was a huge entrepreneurial activity in the Valley, so it came at the right time. And we developed the methodology. It made it easier for people to give back, and if you want to learn about how to start a company in an organized way and network - it's a place. Today it is all over the world. It's got 53 chapters and it's amazing. So it's another-- you have to call it, it's a Silicon Valley phenomenon. We are part of Silicon Valley, yes, having come from that part of the world, but we found a vehicle. This became our vehicle for giving back. And we set in proper governance and now it's what, 17-years-old, and I'm going to say...

Fairbairn: So do you have monthly meetings here, or what's the mechanism?

Patil: There is so much activity. Literally in Silicon Valley there probably are 52 to 70 events every year. There are special groups. There is this. There is that. There are all kinds of things. Then there is the one grand TiEcon the largest conference of the entrepreneurs. So that takes place one week before Mother's Day.

Fairbairn: And do people come from all over the world?

Patil: All over the world. We've got chapters in India, Pakistan, Malaysia, Australia, Singapore, Nordic Country, of course, Great Britain, Canada, of course, and U.S. So first it was U.S. then the rest of the world. But it's amazing. So it's a network of people like us.

Fairbairn: Fascinating. That's a way you've really found a way to give back to the community and to the technical community and help others follow and share the kind of success that you've had. What other things have you taken on to bring your expertise to other activities?

Patil: Well, I've been involved with the Computer History Museum from its very early days. From the time it was in Boston near the harbor and we built the walk through computer. Gordon Bell has been my mentor and I've known him, and he started this thing, and he told me, "Hey, Suhas, you need to get involved," and so I was more than willing to do that. And so I was part of the decision making to bring it to Silicon Valley. And until recently I was a trustee, and because I'm doing a startup right now they've given me some breathing room. They said, "Okay, you do the startup, but you've got to come back." So anyway I have felt people like me joined by a whole bunch of people here the need to sort of have a museum of both its quality for not only what it collects, how it presents, how it reaches, and this work must be done now. And we have these oral histories, we do histories, all kind of things here collected, and people are giving things that otherwise we didn't know had been preserved. So there's a place to bring it. So it will be a place to serve the scholarly purposes, educational purposes, and having been involved with TiE now I understand nonprofits. I actually also was very much a part of when we said, "Okay, we need to have a place," I said, "Yeah, museums don't work unless you have a place," and fortunately this building came up, and so I was part of it. It's a tremendous nice collective effort. And why Silicon Valley, because we have a concentration here of people, so it needs energy of people and financial support of people. So Silicon Valley has this unique situation where there is strength in this thing today. Boston had certain strengths, but really at the moment it is Silicon Valley. So I'm very happy we came here. So I've been involved with The Tech Museum certainly in the early days. I'm not on the board right now. The World Affairs Council , which I was very much involved with earlier.

Fairbairn: And you mentioned that you're involved in a startup so the wheel has turned again. And so just to wrap up the discussion today just give us a feeling of how did that happen and what direction are you off on now.

Patil: Well, after Cirrus Logic's success people would seek me out for advice, and clever entrepreneurs now know how to reach into your pocket. So that's how you become an angel investor, and you do this. This is something I've been doing since early '90s. After Cirrus Logic I was really working with others to make them successful, and this is after many years I'm actually running a company. I'm responsible for payroll and everything. And it's very interesting. The challenges are different, but every company because you're trying to do something on the cutting edge, I don't care which company, very few companies just sail through like that. It is really the skill and the will and the coming-together of people that make these things happen. And I got into it because I told you earlier Cradle Technologies took out this multi-core out of Cirrus Logic. It's a spin off, and got funded. I participated. Not everything goes always well. Technology wise they did fantastic, but the folks who ran it failed to get it into escape velocity. After a lot of support from the VCs finally they gave up. The engineers wouldn't give up. We still believed in it. And since I was the largest investor I lost all my money in multiple washout rounds, but the engineers said, "Well, we'd still like to do that." I know the value of technology including I wanted it to not be lost and to be preserved and because I do believe in it. So I said, "Okay, but maybe nobody will support this thing because the company hasn't succeeded. If you're ready for a rough ride, and you have still got fire in your belly we'll do it." So I actually acquired the company. And a sort of consequence of that is now nobody else is running. I am responsible and I'm working with a smaller team. Even living through this difficult financial period and taking much responsibility on myself supported by some good friends, other entrepreneurs, but primarily me. So we're in the middle of it, and I'm reliving. I can now go back, "Oh, I remember this situation earlier." So situations are not surprising. It doesn't make it any easier. So I'm having to take off. I said, "Oh, I am now in 1984, '85 in the life of Cirrus Logic. So it keeps me energized and I'm working just like I'm a young entrepreneur, you have no choice. Yes, it is hard in some ways, but I tell you, what else would I do? I mean, I've got to be engaged in activity, so that's what I'm doing. And the jury's still out but heck, in Silicon Valley we've still got the chutzpah, we have got the energy, and against all odds we have to succeed. That's what all entrepreneurs, it's all about.

Fairbairn: That's why we're here. We'll, I think on that note is a good time to conclude the discussion and congratulations on your success and being a model in terms of what it takes to be an entrepreneur and the excitement and energy and joy that comes from being that.

Patil: Well, thank you much.

Fairbairn: And also for your contributions to the Computer History Museum, to the community at large through TiE and your personal time, and so thank you, again, for spending the time with us today.

Patil: Thank you much.

END OF INTERVIEW

